**4.感知机**

**4.1定义**

输入为实例的特征向量输出为实例的类别（取+1和-1二值）的二分类线性分类模型。

感知机模型对应于输入空间（特征空间）中将实例划分为正负两类的超平面，是一种线性分类模型，属于判别模型。感知机是神经网络和支持向量机的基础。

假设输入空间（特征空间）是χ⊆Rn，输出空间是У={+1，-1}。输入x∈χ表示实例的特征向量，对应于输入空间（特征空间）的点；输出y∈У表示实例的类别。由输入空间到输出空间的函数如下：

f(x) = sign(w·x + b)

上式即感知机，w∈Rn和b∈R为感知机模型参数，w称为权值或者权值向量，b叫作偏置。sign是符号函数：

优点：简单而易于实现

**4.2感知机学习**

**4.2.1学习策略**

机器学习策略一般都是损失函数最小化原则，损失函数的一个自然选择是误分类点的总数，但是这样的损失函数不是对参数w，b连续可导函数，很难优化。感知机损失函数是误分类点到超平面S的总距离。

给定训练数据集

T = {(x1, y1), (x2, y2), ......, (xN, yN)}

其中xi∈χ=Rn，yi∈У={+1，-1}，i = 1，2，…，N。感知机sign(w·x + b)学习的损失函数定义为

其中M为误分类点的集合，对误分类点(xi，yi)来说yi(w·xi + b) ≤ 0。

由此，感知机学习问题转化为求解损失函数最优化问题：

最优化的方法是**随机梯度下降法**。感知机学习算法可以分为原始形式和对偶形式。

**4.2.2学习算法的原始形式**

感知机学习算法是误分类驱动的，首先任意选取一个超平面w0和b0，然后一次从M中随机选取一个误分类点使其梯度下降，不断极小化损失函数。对损失函数L(w，b)求导得到梯度为

随机选取一个误分类点(xi，yi)，对w，b进行更新

其中η（0 < η ≤ 1）是步长，又称为学习率（learning rate）。

**原始形式算法流程：**

输入：训练数据集T，学习率η；

输出：w，b；感知机模型f(x) = sign(w·x + b)。

（1）选取初值w0，b0；

（2）在训练集中随机选取数据(xi，yi)；

（3）如果yi(w·xi + b) ≤ 0，则

（4）转至（2），直至训练集中没有误分类点。

感知机学习算法可以认为当一个实例点被误分类即位于分离超平面的错误一侧时，调整w，b的值使分离超平面向该误分类点的一侧移动，以减少该误分类点与超平面之间的距离，直至超平面越过该误分类点使其被正确分类。

感知机学习算法对初值敏感，采用不同的初值或选取不同的误分类点，解可以不同。

**4.2.3学习算法的对偶形式**

对偶形式的基本想法是，将w和b表示为实例xi和类标记yi的线性组合的形式，通过求解其系数而求得w和b。

假设原始形式算法中初始值w0和b0均设为0，对误分类点(xi，yi)通过

对w，b逐步修改n次，则w，b关于(xi，yi)的增量分别是αiyixi和αiyi，其中αi=niη（ni表示误分类点(xi，yi)更新的次数）。由此可知最后学习到的w，b可以分别表示为

其中αi≥0，当η=1时，表示第i个实例因为误分类而更新的次数，实例点更新次数越多，意味着它距离分离超平面越近，也就越难正确分类，即这样的实例对学习结果影响最大。

**对偶形式算法流程：**

输入：线性可分数据集T，学习率η；

输出：a，b；感知机模型

其中α = (α1，α2，…，αN)T。

（1）α←0，b←0；

（2）在训练集中随机选取数据(xi，yi)；

（3）如果yi ≤ 0，则

αi←αi + η

b←b + ηyi

（4）转至（2），直至训练集中没有误分类点。

对偶形式训练实例仅以内积形式出现，可预先将训练集实例间的内积计算出来以矩阵的形式存储，即Gram矩阵（Gram matrix）：G = [xi·xj]N\*N。

**附录：算法的收敛性**

**1.数据集线性可分性**

给定训练数据集

T = {(x1, y1), (x2, y2), ......, (xN, yN)}

其中xi∈χ=Rn，yi∈У={+1，-1}，i = 1，2，…，N。如果存在某个超平面S

w·x + b=0

能够将数据集的正负实例点完全正确地划分到超平面的两侧，即对所有yi =+1的实例i，有w·xi + b > 0，对所有的yi =-1的实例i，有w·xi + b < 0，则称数据集T为线性可分数据集（linearly separable data set）；否则称数据集T线性不可分。

**2.算法的收敛性**